



## PATENT ABSTRACTS OF JAPAN

(11) Publication number: **10210066 A**(43) Date of publication of application: **07 . 08 . 98**

(51) Int. Cl. **H04L 12/46**  
**H04L 12/28**  
**G06F 12/00**  
**G06F 13/00**

(21) Application number: **09005880**(71) Applicant: **SUMITOMO ELECTRIC IND LTD**(22) Date of filing: **16 . 01 . 97**(72) Inventor: **YAMASHITA KAZUHIKA**

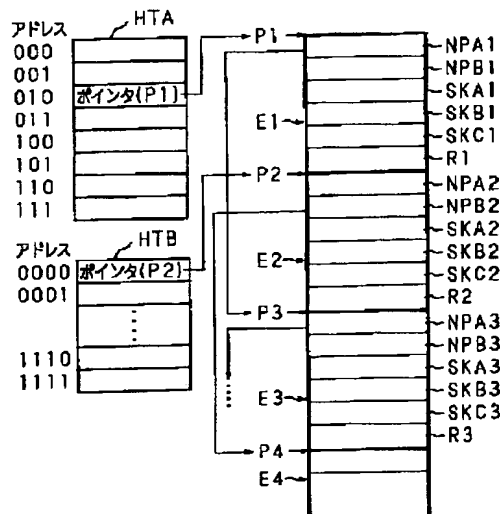
(54) **DATA BASE FOR JUDGING INTERNET WORK  
 REPEATING DESTINATION AND METHOD FOR  
 CONSTRUCTING DATA BASE**

## (57) Abstract:

**PROBLEM TO BE SOLVED:** To reduce the number of entries by using different hash tables at the time of learning and at the time of actually judging the repeating destination, and allowing each entry to have a pointer for the both hash tables.

**SOLUTION:** A first hash table HTA is used at the time of learning and a second hash table HTB is used at the time of actually searching the repeating destination in a data base for judging the internet work repeating. The pointer of each entry is registered in the both hash tables HTA and HTB by using a hash value as an index. Therefore, when a pointer (for example, P1) is registered at the position of the first hash table HTA corresponding to a hash value '010' of a certain MAC address, the entry designated, that is, linked by the pointer (P1) (for example, an entry E) is referred to at first.

COPYRIGHT: (C)1998,JPO

**BEST AVAILABLE COPY**

(19)日本国特許庁 (J P)

(12) 公開特許公報 (A)

(11)特許出願公開番号

特開平10-210066

(43)公開日 平成10年(1998) 8月7日

(51)Int.Cl. <sup>6</sup>	識別記号	F I
H 0 4 L 12/46		H 0 4 L 11/00
12/28		G 0 6 F 12/00
G 0 6 F 12/00	5 2 0	13/00
13/00	3 5 5	3 1 0 C
		5 2 0 A
		3 5 5

審査請求 未請求 請求項の数7 O L (全 18 頁)

(21)出願番号 特願平9-5880

(22)出願日 平成9年(1997) 1月16日

(71)出願人 000002130

住友電気工業株式会社

大阪府大阪市中央区北浜四丁目5番33号

(72)発明者 山下 和寿

大阪府大阪市此花区島屋一丁目1番3号

住友電気工業株式会社大阪製作所内

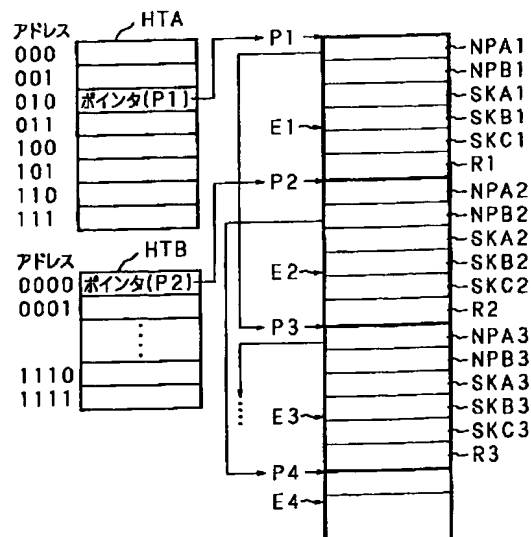
(74)代理人 弁理士 河野 登夫

(54)【発明の名称】 ネットワーク間中継先判定用データベース及びデータベースの構築方法

## (57)【要約】

【課題】 従来は多数のサーチキーが同一のハッシュ値に変換されるため、目的のエントリを見つけるまでにネクストポイントでチェイニングされた多数のエントリをサーチする必要が生じ、ハッシュ関数を利用しているにも拘らず効率的なサーチが行なえないという問題が生じる。

【解決手段】 ハッシュ関数を使用することにより本来のサーチキーをサーチするために必要な空間を圧縮することは従来と同様であるが、学習時と実際の中継先判定時とで異なるハッシュテーブルHTA, HTBを使用し、しかも各エントリE1, E2, E3…に両方のハッシュテーブルHTA, HTB用の第1ネクストポイントNPA, 第2ネクストポイントNPBを持たせておくことにより、両ハッシュテーブルHTA, HTBにリンクするエントリを共通化して、エントリ数を従来のハッシュテーブル使用の場合と同等に抑えることを特徴とする。



NPA I : 第1ネクスト

ポイント

NPB I : 第2ネクスト

ポイント

SKA I : 第1サーチキー

SKB I : 第2サーチキー

SKC I : 第3サーチキー

## 【特許請求の範囲】

【請求項1】 それぞれにアドレスが付与された複数の端末を接続したネットワーク複数がそれぞれ中継装置の異なるポートを介して接続されたネットワーク間で、通信の送信元の端末を示すアドレスと通信の送信先の端末を示すアドレスとを含む通信信号を通信する際の前記中継装置における中継先判定用データベースにおいて、端末のアドレスと、それぞれが前記中継装置に接続されるポートのアドレスと、それぞれが属するネットワークのアドレスとを保持するエン트리と、

端末のアドレスとそれが前記中継装置に接続されるポートのアドレスとを変数として第1のハッシュ関数により求めた値をインデックスとする第1のハッシュテーブルと、

端末のアドレスとその端末が属するネットワークのアドレスとを変数として第2のハッシュ関数により求めた値をインデックスとする第2のハッシュテーブルとを有し、前記第1のハッシュテーブルの各インデックスに対応する位置には、端末のアドレスとポートのアドレスとが前記第1のハッシュ関数によりそれぞれのインデックスに変換される一つまたは複数のエントリの内の一つにリンクするためのポインタを、前記第2のハッシュテーブルの各インデックスに対応する位置には、端末のアドレスとネットワークのアドレスとが前記第2のハッシュ関数によりそれぞれのインデックスに変換される一つまたは複数のエントリの内の一つにリンクするためのポインタをそれぞれ保持させてあり、

各エン 트리には、前記第1のハッシュテーブルの同一のインデックスに前記第1のハッシュ関数により変換されるべき端末のアドレスとポートのアドレスとを有する複数のエン 트리相互を順次的にリンクさせるための第1のポインタと、前記第2のハッシュテーブルの同一のインデックスに前記第2のハッシュ関数により変換されるべき端末のアドレスとネットワークのアドレスとを有する複数のエン 트리相互を順次的にリンクさせるための第2のポインタとをそれぞれ保持させてあることを特徴とするネットワーク間中継先判定用データベース。

【請求項2】 前記第1のハッシュ関数（又は第2のハッシュ関数）により前記第1のポインタ（又は第2のポインタ）は、前記第1のハッシュテーブル（又は第2のハッシュテーブル）の同一のインデックスに変換されるべき端末のアドレスとポートのアドレス（又は端末のアドレスとネットワークのアドレス）とを有する複数のエント리를、端末のアドレスとポートのアドレス（又は端末のアドレスとネットワークのアドレス）とにより一意に決定される値の順序に従ってリンクするように設定されていることを特徴とする請求項1に記載のネットワーク間中継先判定用データベース。

【請求項3】 前記第1のハッシュ関数（又は第2のハッシュ関数）により前記第1のポインタ（又は第2のポ

インタ）は、前記第1のハッシュテーブル（又は第2のハッシュテーブル）の同一のインデックスに変換されるべき端末のアドレスとポートのアドレス（又は端末のアドレスとネットワークのアドレス）とを有する複数のエント리를、端末のアドレスとポートのアドレス（又は端末のアドレスとネットワークのアドレス）とにより一意に決定される値の昇順及び降順にリンクする二種類のポインタを有することを特徴とする請求項1に記載のネットワーク間中継先判定用データベース。

10 【請求項4】 前記第1のハッシュ関数（又は第2のハッシュ関数）により前記第1のハッシュテーブル（又は第2のハッシュテーブル）の同一のインデックスに変換されるべき端末のアドレスとポートのアドレス（又は端末のアドレスとネットワークのアドレス）とにより一意に決定される値の中間の値が前記第1のハッシュテーブル（又は第2のハッシュテーブル）の対応する位置に登録されており、

前記第1のポインタ（又は第2のポインタ）は、前記第1のハッシュテーブル（又は第2のハッシュテーブル）の同一のインデックスに前記第1のハッシュ関数（又は第2のハッシュ関数）により変換されるべき端末のアドレスとポートのアドレス（又は端末のアドレスとネットワークのアドレス）とを有する複数のエントりに、前記第1のハッシュテーブル（又は第2のハッシュテーブル）の対応するインデックスの位置に登録されている値を起点として、端末のアドレスとポートのアドレス（又は端末のアドレスとネットワークのアドレス）とにより一意に決定される値の昇順及び降順にリンクする二種類のポインタを有することを特徴とする請求項1に記載のネットワーク間中継先判定用データベース。

30 【請求項5】 サーチキーを変数としてハッシュ関数からハッシュ値を求め、求められたハッシュ値に対応するハッシュテーブルのインデックスの位置に前記サーチキーに対応するデータが登録されたエント리를リンクさせるデータベースの構築方法において、

前記ハッシュ関数により前記ハッシュテーブルの同一のインデックスに変換されるべきサーチキーを有する複数のエント리를、それらの内のサーチキーにより一意に決定される値が最大または最小の一つがハッシュテーブルの対応するインデックスの位置にリンクされ、他のそれぞれがサーチキーにより一意に決定される値の順序に従ってリンクすることを特徴とするデータベースの構築方法。

40 【請求項6】 サーチキーを変数としてハッシュ関数からハッシュ値を求め、求められたハッシュ値に対応するハッシュテーブルのインデックスの位置に前記サーチキーに対応するデータが登録されたエント리를リンクさせるデータベースの構築方法において、

前記ハッシュ関数により前記ハッシュテーブルの同一のインデックスに変換されるべきサーチキーを有する複数のエント리를、それらの内の一つが前記ハッシュテーブル

の対応するインデクスの位置にリンクされ、他のそれぞれがサーチキーにより一意に決定される値の昇順及び降順にリンクすることを特徴とするデータベースの構築方法。

【請求項7】 サーチキーを変数としてハッシュ関数からハッシュ値を求め、求められたハッシュ値に対応するハッシュテーブルのインデクスの位置に前記サーチキーに対応するデータが登録されたエントリをリンクさせるデータベースの構築方法において、

前記ハッシュ関数により前記ハッシュテーブルの同一のインデクスに変換されるべきサーチキーにより一意に決定される値の中間の値を前記ハッシュテーブルの対応する位置に登録し、

前記ハッシュ関数により前記ハッシュテーブルの同一のインデクスに変換されるべきサーチキーを有する複数のエントリを、それらの内の一つが前記ハッシュテーブルの対応するインデクスの位置にリンクされ、他のそれぞれが前記ハッシュテーブルの対応するインデクスの位置に登録されている値を起点として、サーチキーにより一意に決定される値の昇順及び降順にリンクすることを特徴とするデータベースの構築方法。

【発明の詳細な説明】

【0001】

【発明の属する技術分野】本発明はネットワーク中継装置、具体的にはLAN(Local Area Network)を複数接続したネットワーク相互を接続したネットワーク間中継装置におけるネットワーク間中継先判定用データベースに関する。また、そのような用途に使用されるデータベースの構築方法に関する。

【0002】

【従来の技術】LANは一般的には構内情報通信網等と称され、通常は公衆通信回線を経由せずに接続される範囲内のネットワークである。たとえば、一つの建物内、一つのフロア内、一つの工場の敷地内等のような比較的小範囲に分散している複数のコンピュータシステム間において、情報の共有、周辺機器（プリンタ等）の共有等を目的とする。

【0003】このようなLANも、近年のコンピュータ通信の発展に伴って、複数のLANを相互に接続することが求められており、そのような複数のLANが接続されたネットワークをWAN(Wide Area Network)と称する。ところで、一つのLANが孤立している場合にはそれに接続するコンピュータシステムも比較的少数であり、またコンピュータ間の伝送路も単純であるため、その孤立しているLAN内でのコンピュータ間通信に際してもそれほど困難はなかった。しかし、複数のLANが相互に接続されたWANにおいては、コンピュータ通信を必要な宛先に正確且つ迅速に届けるためには、宛先の指定方法及び通信経路の選択が重要な課題となる。

【0004】通信経路の選択に関しては、LAN相互の接

続点に位置する中継装置の機能に負うところが大きい。

LAN間接続のための中継装置には、低機能側から順に、レピータ、ブリッジ、ルータ、ゲートウェイ等の種々の装置がある。レピータは単なる物理的な接続を行なうのみであるが、他は通信信号の送信先に応じてその信号を通過させたり、あるいは通過させなかつたり等の処理を行なうことが可能である。図1の模式図にブリッジを使用したLAN間接続の例を示す。

【0005】この例では、リング型のLAN Xと、同じくリング型のLAN Zと、バス型のLAN YとがブリッジBRにより相互に接続されている。より具体的には、LAN XはブリッジBRの第1ポートPT1に、LAN YはブリッジBRの第2ポートPT2に、LAN ZはブリッジBRの第3ポートPT3にそれぞれ接続している。TA, TB, TC…はそれぞれパーソナルコンピュータ、ワークステーション等の端末を示している。なお、端末THは物理的にはLAN Zに接続しているが、ソフトウェア的な処理により仮想的にLAN Xの一部と見做される。従って、本来のLAN X（この例では端末TE, TF, TG, TI等が接続している）と端末THとでV-LAN(バーチャルLAN) Xを構成する。換言すれば、V-LAN XはブリッジBRに対して第1ポートPT1及び第3ポートPT3の二つのポートで接続されていることになる。

【0006】このようなLAN間ネットワークにおいて、たとえばV-LAN Xに属する端末TFからLAN Yに属する端末TAへ送信する場合には、図2の模式図に示されているようなフォーマットの信号が送信される。このフォーマットは、デスティネーションアドレス(DA)のフィールドF1と、ソースアドレス(SA)のフィールドF2と、本来の通信情報のフィールドF3とで構成されている。デスティネーションアドレス(DA)とはその通信の送信先の端末のアドレスであり、ソースアドレス(SA)とはその通信の送信元のアドレスである。なお、デスティネーションアドレス(DA)のフィールドF1とソースアドレス(SA)のフィールドF2とをまとめてヘッダと称する。

【0007】このようなフォーマットの信号がV-LAN Xの端末TFからブリッジBRへその第1ポートPT1経由で到達すると、ブリッジBRはヘッダの内容を調べてその信号をいずれのポートへ送出するかを判断を行なう。この例では、V-LAN Xに属する端末TFからLAN Yに属する端末TAへの送信であるから、ブリッジBRはこの信号を第2ポートPT2へ送出しなければならない。そのようなブリッジBRによるポート選択の判断は予め用意されているフィルタリングデータベースと称されるデータベースをヘッダの内容をサーチキーとしてサーチすることにより行なわれるが、ヘッダの内容であるデスティネーションアドレス(DA)及びソースアドレス(SA)には通常はMAC(Media Access Control)アドレスと称されるアドレスが使用される。なお、上述のデータベースは具体的には、ブリッジBRと接続されたハードディスク、RAM等の書換え可能な記録媒体RMに電子的な情報として記録されている。

【0008】ところで、この MACアドレスは48ビットのバイナリデータであり、従って $2^{48}$ (約 280兆) 通りの膨大なアドレス空間を表現することが可能である。しかし、現実には LANあるいはWAN に接続する端末の数はそれほどではないため、MACアドレス自体もせいぜい数千アドレス程度の空間の表現が可能であればよい。一方、接続装置(ブリッジBR)の側においても、そのような48ビットものアドレス空間を管理するためには $2^{48}$ 個のエントリを有するテーブルを用意する必要がある。このことはそのテーブルの大きさに見合ったメモリをハードウェアとして備える必要があることを意味しており、この面からも非現実的である。

【0009】このような事情から、従来はハッシュ関数と称される関数を利用して MACアドレスの空間を圧縮する手法が採られていた。ハッシュ関数とはたとえば、住所録を作成する場合に、人名の文字コードを加算した数値の下位側の幾つかの桁の値に各人の住所を対応付けて登録するような手法である。このようにして作成されたテーブルでは、ハッシュ関数から得られたインデックス値(ハッシュ値)を $n$ 、テーブルの先頭アドレスを $a$ 、一つのデータのサイズを $s$ とすると、求めるデータのアドレスは、

$$a + s \times (n - 1)$$

で求まるため、通常必要なテーブルのサーチを行なうことなしに目的のデータをインデックス値から直接参照することが出来る。

【0010】以下、従来のフィルタリングデータベースに使用されていたハッシュ関数の事例について具体的に説明する。いまたとえば、 $X$ を4ビットの数値とすると、 $X$ がとり得る値の範囲は"0000"から"1111"までの範囲である。ここで、ハッシュ関数 $H(X)$ を「 $X$ の下位2ビットのみを表示する関数」と定義すると、たとえば $X = "0000"$ である場合のハッシュ関数 $H(X)$ の値(以下、ハッシュ値と言う)は"00"に、 $X = "0001"$ である場合のハッシュ値は"01"に、 $X = "0110"$ である場合のハッシュ値は"10"に、 $X = "1111"$ である場合のハッシュ値は"11"にそれぞれなる。一方、こられらのハッシュ値をインデックスとしてエントリを関連付けたテーブル(以下、ハッシュテーブルと言う)を作成しておく。以上により、このハッシュテーブルからはハッシュ値をインデックスとして目的のエントリを直接参照することが可能になる。

【0011】このように、ハッシュ関数を利用することにより、本来サーチすべき空間をより小さな空間に圧縮し、しかもその小さな空間を本来のサーチキーをハッシュ関数で変換したハッシュ値をインデックスとして直接参照出来るため、1回の参照で目的とするエントリを見つけ、そこから必要なデータを直ちに得ることが出来る可能性も高くなる。なお、サーチキーのハッシュ関数によ

るハッシュ値(インデックス)への変換に際して、複数のサーチキー(MACアドレス)が同一のハッシュ値に変換される場合(上述の場合であればたとえば $X = "0000"$ と"1100"とはハッシュ値は共に"00"になる)には同一のインデックスにおいて複数のサーチキーに対応するエントリが衝突することになるが、その場合には衝突したエントリをたとえばチェイニングまたはオープンアドレッシングにより関連付けて参照可能にする。

【0012】オープンアドレッシングとは、エントリが衝突した場合には他の未使用のアドレスに衝突した一方のエントリを登録する手法である。チェイニングとは、衝突した複数のエントリをリンクドリストで関連付けておき、逐次的に参照してゆくことにより、必要なエントリをサーチする手法である。

【0013】ところで、上述のようなハッシュ関数を利用するしないには拘らず、フィルタリングデータベースを使用するには予め情報が登録されている必要がある。換言すれば、ブリッジBRは各端末へ送信を行なう場合に自身のポートの内のいずれから送出すればよいかを前もって学習しておく必要がある。この学習は、たとえばある送信フレームがブリッジBRのあるポートから入力された場合に、そのフレームのヘッダに含まれるソースアドレス(SA)とそのフレームを受信したポートアドレスとを一組のレコードとし、ソースアドレス(SA)のハッシュ値をインデックスとしてフィルタリングデータベース(ハッシュテーブル)に登録することにより行なわれる。なお、実際には後述するように、ハッシュテーブルそのものにはレコードが直接登録されるのではなく、そのレコードが登録されたエントリのアドレスを示すポインタが登録される。

【0014】上述のような登録が一旦行なわれた後は、その MACアドレスをデスティネーションアドレス(DA)とする送信フレームをブリッジBRが受信した場合にそのフレームをいずれのポートへ送出すればよいかは、そのフレームのデスティネーションアドレス(DA)をサーチキーとしてハッシュ値を求め、それをインデックスとしてハッシュテーブルを参照すればよいことになる。

【0015】また、上述のような場合に、複数の端末の MACアドレス、即ちサーチキーが同一のハッシュ値を採る可能性は常にあり得る。複数の端末の MACアドレスが同一のハッシュ値をとる場合、即ち前述のチェイニングによるリスト構造が必要になる場合について説明する。

【0016】図3は従来の一般的なハッシュテーブル及び個々のエントリの関係を示す模式図である。図3において、参照符号HTはハッシュテーブルを、E1, E2, E3…はそれぞれエントリを示している。なお、各エントリE1, E2, E3…は具体的にはデータベースとして構築されており、同一のハッシュ値にリンクされる他のエントリをリンクさせるためのネクストポインタNP1, NP2, NP3…と、そのエントリを特定するための本来の MACアドレ

スであるサーチキーSK1, SK2, SK3 …と、付属情報R1, R2, R3…等とを有する。

【0017】ハッシュテーブルHTには、個々のエントリのポインタが、ハッシュ値をインデクスとして登録されている。従って、図3に示されているように、たとえばハッシュテーブルHTのあるMACアドレスのハッシュ値に対応する位置にポインタ（たとえばP1）が登録されている場合には、このポインタP1で指定されるエントリ（たとえばE1）がまず最初に参照される。しかし、前述のように、複数のMACアドレスが同一のハッシュ値に変換された場合には、そのハッシュ値において二つ以上のMACアドレスが衝突するため、他のエントリをリンクさせるためのネクストポインタNPが登録されている。また、ネクストポインタにより複数のエントリがリンクされている場合にはどのエントリがどのMACアドレス（サーチキー）に本来対応しているかを識別するために本来のMACアドレスも記録されている。

【0018】たとえば図3に示されている例では、ハッシュテーブルHTのあるMACアドレス（サーチキー）から生成されたハッシュ値に対応するインデクスの位置にポインタP1が記録されているとすると、それに対応するエントリE1が参照される。このエントリE1には更にネクストポインタNP1としてP3が、またサーチキーSK1としてX1がそれぞれ記録されている。従って、ハッシュ値が作成された元のMACアドレス（サーチキー）がX1であればエントリE1が目的のエントリである。

【0019】しかし、元のMACアドレス（サーチキー）がX1ではなかった場合にはネクストポインタNP1に登録されているポインタ値P3でリンクされているエントリE3が次に参照される。このエントリE3に登録されているMACアドレスX5が元のMACアドレス（サーチキー）であった場合にはこのエントリE3が参照される。しかし、エントリE3に登録されているMACアドレスも元のMACアドレス（サーチキー）と異なる場合には更にそのエントリE3のネクストポインタNP3に登録されているポインタ値P6でリンクされているエントリE6が次に参照される。

【0020】このようにして、エントリが順次的にリンクされているチェイニングリストが目的のエントリが見つかるまで調べられるが、ネクストポインタNPに次のポインタ値が登録されていない場合にはそこでエントリのリンクが終了していることを意味している。従って、チェイニングリストにリストされているエントリを順にたどっていても目的のエントリが見つからないままにリストが終了した場合には、対応するエントリが未だ登録されていないことを意味する。換言すれば、そのような場合にはMACアドレスに対応するエントリを新たに登録する必要がある。

【0021】次に、前述の図1に示されているようなネットワークにおいて、ブリッジBRがハッシュテーブルを使用して送信フレームの送出先のポートをサーチする場

合について説明する。前述の従来の説明に用いた図1において、ブリッジBRの第1ポートPT1にアドレス〔0000〕を、第2ポートPT2にアドレス〔0001〕を、第3ポートPT3にアドレス〔0010〕をそれぞれ付与する。そして、各端末には図4に示されているような48ビットのMACアドレスが付与されているとする。

【0022】ブリッジBRには各端末に関する情報として、それぞれのMACアドレスと、ブリッジBRから見てそれぞれが接続されているポートアドレスとが与えられる。ここで、各端末に関して、MACアドレスをサーチキーとし、ハッシュ関数H(X)としてMACアドレスの下位3ビットを採用する。この結果、それぞれの端末に関するハッシュ関数H(X)は図4の模式図に示されているようになる。従って、本来は48ビットのMACアドレスから3ビットのハッシュ関数H(X)が作成される。ハッシュ関数H(X)は3ビットであるため8通りの値しかとらない。

【0023】いまたたとえば、MACアドレスが〔・・・・010〕である端末TFからMACアドレスが〔・・・・000〕である端末TAへ通信を行なう場合について考える。この場合の送信フレームは図2の模式図に示されているヘッダ内のフィールド1にはデスティネーションアドレス(DA)として端末TAのMACアドレス〔・・・・0000〕が、フィールド2にはソースアドレス(SA)として端末TFのMACアドレス〔・・・・0010〕が入っている。またこの送信フレームは、端末TFから送出されて第1ポート（ポートアドレス〔0000〕）からブリッジBRに入力する。

【0024】このようにブリッジBRに端末TFからの送信フレームが入力すると、まずサーチキーであるソースアドレスSA〔・・・・0010〕からハッシュ関数H(X)によりハッシュ値“010”が求められる。

【0025】ハッシュ値が求まるとそれをインデクスとしてハッシュテーブルHTがサーチされる。ハッシュテーブルHTの対応する位置にポインタが既に登録されていればそのポインタが読み出され、そのポインタにリンクされているエントリが参照され、それに登録されているサーチキー(MACアドレス)が読み出される。このMACアドレスが本来のサーチキーと一致していればサーチはここで終了し、そのエントリからポートアドレスが取り出され、実際に送信フレームが入力されたポートと一致しているかが確認され、すでに登録されているポートアドレスが今回送信フレームが入力したポートのアドレスと一致していなければ今回のポートアドレスに更新され、一致していれば更新する必要はない。また同時に、付属情報R内の更新履歴が更新される。

【0026】しかし、上述のようにして参照されたエントリが本来のサーチキーに対応するものではなかった場合にはそのエントリに登録されているネクストポインタでリンクされているエントリが次に参照される。以下同様にして、目的のエントリ、即ち本来のサーチキーに対

応するエントリが見つかるまで、チェイニングリストの各エントリが次々にサーチされる。

【0027】ところで、上述の例において、ハッシュテーブルHTのインデクス”010”に対応する位置にポインタが無かった場合、及び上述のようにしてリンクされているエントリが次々にサーチされても本来のサーチキーに対応するエントリが見つからなかった場合には、サーチキーに対応するエントリが未だ登録されていないことになる。この場合には、送信フレームの内容に従ってハッシュテーブルHTのインデクス”010”に対応する位置にポインタを新たに登録すると共にそのポインタにリンクさせたエントリを新たに追加し、必要な情報を登録し、更新履歴として新規登録の履歴を登録しておく。以上が学習時の処理である。

【0028】実際の中継先判定の処理は以下になる。送信フレームのデスティネーションアドレスDAをサーチキーとし、ハッシュテーブルHTをサーチする。具体的には、サーチキーである端末TAのMACアドレス、即ちデスティネーションアドレスDA[・・・・0000]からハッシュ関数H(X)に従ってその下位3ビット”000”がハッシュ値とされる。

【0029】ハッシュ値が求まるとそれをインデクスとしてハッシュテーブルHTがサーチされる。インデクス”000”に対応するハッシュテーブルHTの位置にポインタが既に登録されていれば、それにリンクしているエントリが参照され、それからポートアドレスを取り出し、そのポートを送信フレームの送出先のポートと見做す。

【0030】しかし、上述のようにして参照されたエントリが本来のサーチキーに対応するものではなかった場合にはそのエントリに登録されているネクストポインタでリンクされているエントリが次に参照される。以下同様にして、目的のエントリ、即ち本来のサーチキーに対応するエントリが見つかるまで、チェイニングリストの各エントリが次々にサーチされる。

【0031】ところで、ハッシュテーブルHTのインデクス”000”に対応する位置にポインタが無かった場合、及び上述のようにしてチェイニングリストの各エントリが次々にサーチされても本来のサーチキーに対応するエントリが見つからなかった場合には、サーチキーに対応するエントリが未だ登録されていないことになる。この場合には、ブリッジBRが有するすべてのポート（但し、受信ポート、この場合はポートアドレス[0000]の第1ポートは除く）を送出先ポートと見做す。

【0032】

【発明が解決しようとする課題】しかし上述のような従来の手法では、多数のサーチキーが同一のハッシュ値に変換されるため、目的のエントリを見つけるまでにネクストポインタでチェイニングされた多数のエントリをサーチする必要が生じ、ハッシュ関数を利用しているにも拘らず効率的なサーチが行なえないという問題が生じ

る。この場合、たとえば学習時のハッシュテーブルと実際の中継先判定時のハッシュテーブルとを別個にすることが考えられるが、そうするとそれぞれのハッシュテーブルにリンクされたエントリを作成する必要が生じ、エントリのためのメモリ容量が単純に考えても2倍必要になる。

【0033】本発明はこのような事情に鑑みてなされたものであり、ハッシュ関数を使用することにより本来のサーチキーをサーチするために必要な空間を圧縮することは従来と同様であるが、エントリの数を従来のハッシュテーブル使用の場合と同等に抑えたままでより効率的なサーチを可能として、上述のような問題点の解決を図ったネットワーク間中継装置の中継先判定方法及びそのような用途に使用されるデータベースの構築方法の提供を目的とする。

【0034】

【課題を解決するための手段】本発明に係るネットワーク間中継先判定用データベース及びデータベースの構築方法は、端的には、ハッシュ関数を使用することにより本来のサーチキーをサーチするために必要な空間を圧縮することは従来と同様であるが、学習時と実際の中継先判定時とで異なるハッシュテーブルを使用し、しかも各エントリに両方のハッシュテーブル用のポインタを持たせておくことにより、両ハッシュテーブルにリンクするエントリを共通化することにより、エントリの数を従来のハッシュテーブル使用の場合と同等に抑えることを特徴とする。

【0035】また、各エントリのリンクの順序を工夫することにより、目的のエントリを迅速にサーチし、また目的のエントリが未登録であることを迅速に検出し得るようにした。

【0036】本発明に係るネットワーク間中継先判定用データベースは、それぞれにアドレスが付与された複数の端末を接続したネットワーク複数がそれぞれ中継装置の異なるポートを介して接続されたネットワーク間で、通信の送信元の端末を示すアドレスと通信の送信先の端末を示すアドレスとを含む通信信号を通信する際の中継装置における中継先判定用データベースであって、端末のアドレスと、それぞれが中継装置に接続されるポートのアドレスと、それぞれが属するネットワークのアドレスとを保持するエントリと、端末のアドレスとそれが中継装置に接続されるポートのアドレスとを変数として第1のハッシュ関数により求めた値をインデクスとする第1のハッシュテーブルと、端末のアドレスとその端末が属するネットワークのアドレスとを変数として第2のハッシュ関数により求めた値をインデクスとする第2のハッシュテーブルとを有し、第1のハッシュテーブルの各インデクスに対応する位置には、端末のアドレスとポートのアドレスとが第1のハッシュ関数によりそれぞれのインデクスに変換される一つまたは複数のエントリの内

の一つにリンクするためのポインタを、第2のハッシュテーブルの各インデクスに対応する位置には、端末のアドレスとネットワークのアドレスとが第2のハッシュ関数によりそれぞれのインデクスに変換される一つまたは複数のエントリの内の一つにリンクするためのポインタをそれぞれ保持させてあり、各エントリには、第1のハッシュテーブルの同一のインデクスに第1のハッシュ関数により変換されるべき端末のアドレスとポートのアドレスとを有する複数のエントリ相互を順次的にリンクさせるための第1のポインタと、第2のハッシュテーブルの同一のインデクスに第2のハッシュ関数により変換されるべき端末のアドレスとネットワークのアドレスとを有する複数のエントリ相互を順次的にリンクさせるための第2のポインタとをそれぞれ保持させてあることを特徴とする。

【0037】また本発明に係るネットワーク間中継先判定用データベースは、上述の構成において、第1のハッシュ関数（又は第2のハッシュ関数）により第1のポインタ（又は第2のポインタ）は、第1のハッシュテーブル（又は第2のハッシュテーブル）の同一のインデクスに変換されるべき端末のアドレスとポートのアドレス（又は端末のアドレスとネットワークのアドレス）とを有する複数のエントリを、端末のアドレスとポートのアドレス（又は端末のアドレスとネットワークのアドレス）とにより一意に決定される値の順序に従ってリンクするように設定されていることを特徴とする。

【0038】更に本発明に係るネットワーク間中継先判定用データベースは、上述の構成において、第1のハッシュ関数（又は第2のハッシュ関数）により第1のポインタ（又は第2のポインタ）は、第1のハッシュテーブル（又は第2のハッシュテーブル）の同一のインデクスに変換されるべき端末のアドレスとポートのアドレス（又は端末のアドレスとネットワークのアドレス）とを有する複数のエントリを、端末のアドレスとポートのアドレス（又は端末のアドレスとネットワークのアドレス）とにより一意に決定される値の昇順及び降順にリンクする二種類のポインタを有することを特徴とする。

【0039】また更に本発明に係るネットワーク間中継先判定用データベースは、上述の構成において、第1のハッシュ関数（又は第2のハッシュ関数）により第1のハッシュテーブル（又は第2のハッシュテーブル）の同一のインデクスに変換されるべき端末のアドレスとポートのアドレス（又は端末のアドレスとネットワークのアドレス）とにより一意に決定される値の中間の値が第1のハッシュテーブル（又は第2のハッシュテーブル）の対応する位置に登録されており、第1のポインタ（又は第2のポインタ）は、第1のハッシュテーブル（又は第2のハッシュテーブル）の同一のインデクスに第1のハッシュ関数（又は第2のハッシュ関数）により変換されるべき端末のアドレスとポートのアドレス（又は端末の

アドレスとネットワークのアドレス）とを有する複数のエントリに、第1のハッシュテーブル（又は第2のハッシュテーブル）の対応するインデクスの位置に登録されている値を起点として、端末のアドレスとポートのアドレス（又は端末のアドレスとネットワークのアドレス）とにより一意に決定される値の昇順及び降順にリンクする二種類のポインタを有することを特徴とする。

【0040】また、本発明に係るデータベースの構築方法は、サーチキーを変数としてハッシュ関数からハッシュ値を求め、求められたハッシュ値に対応するハッシュテーブルのインデクスの位置にサーチキーに対応するデータが登録されたエントリをリンクさせるデータベースの構築方法であって、ハッシュ関数によりハッシュテーブルの同一のインデクスに変換されるべきサーチキーを有する複数のエントリを、それらの内のサーチキーにより一意に決定される値が最大または最小の一つがハッシュテーブルの対応するインデクスの位置にリンクされ、他のそれぞれがサーチキーにより一意に決定される値の順序に従ってリンクすることを特徴とする。

【0041】また本発明に係るデータベースの構築方法は、サーチキーを変数としてハッシュ関数からハッシュ値を求め、求められたハッシュ値に対応するハッシュテーブルのインデクスの位置にサーチキーに対応するデータが登録されたエントリをリンクさせるデータベースの構築方法において、ハッシュ関数によりハッシュテーブルの同一のインデクスに変換されるべきサーチキーを有する複数のエントリを、それらの内の一つがハッシュテーブルの対応するインデクスの位置にリンクされ、他のそれぞれがサーチキーにより一意に決定される値の昇順及び降順にリンクすることを特徴とする。

【0042】更に本発明に係るデータベースの構築方法は、サーチキーを変数としてハッシュ関数からハッシュ値を求め、求められたハッシュ値に対応するハッシュテーブルのインデクスの位置にサーチキーに対応するデータが登録されたエントリをリンクさせるデータベースの構築方法であって、ハッシュ関数によりハッシュテーブルの同一のインデクスに変換されるべきサーチキーにより一意に決定される値の中間の値をハッシュテーブルの対応する位置に登録し、ハッシュ関数によりハッシュテーブルの同一のインデクスに変換されるべきサーチキーを有する複数のエントリを、それらの内の一つがハッシュテーブルの対応するインデクスの位置にリンクされ、他のそれぞれがハッシュテーブルの対応するインデクスの位置に登録されている値を起点として、サーチキーにより一意に決定される値の昇順及び降順にリンクすることを特徴とする。

【0043】

【発明の実施の形態】以下、本発明をその実施の形態を示す図面に基づいて詳述する。前述の従来の説明に用いた図5に示されているように、LAN Xに接続する全ての



端末TE, TF, TG, TI…と、LAN Yの端末TAと、LAN Zの端末THとをバーチャルLAN A (V-LANA) としてアドレス〔0000〕を付与し、LAN Yの端末TA以外の全ての端末TB, TC…と、LAN Zの端末TH以外の全ての端末TJ等とをバーチャルLAN B (V-LANB)としてアドレス〔0010〕を付与する。また、ブリッジBRの第1ポートPT1にアドレス〔0000〕を、第2ポートPT2にアドレス〔0001〕を、第3ポートPT3にアドレス〔0010〕をそれぞれ付与する。そして、各端末には前述の図4に示されている48ビットのMACアドレスが付与されている。なお、本発明のデータベースは具体的には、ブリッジBRと接続されたハードディスク、RAM等の書換え可能な記録媒体RMに電子的な情報として記録されている。

【0044】図6は本発明のネットワーク間中継先判定用データベース及びデータベースの構築方法において個々のエントリの内容を示す模式図である。このエントリは、第1ネクストポイントNPAの欄と、第2ネクストポイントNPBの欄と、第1サーチキーSKAの欄と、第2サーチキーSKBの欄と、第3サーチキーSKCの欄と、たとえば更新履歴等の付属情報Rの欄とを有する。

【0045】本発明においては、各エントリがチェインリストを形成するためのネクストポイントを第1(NPA)、第2(NPB)の2種類有していることが特徴的である。またサーチキーはテーブルを参照するためのキーであり、学習時には第1サーチキーSKAと第2サーチキーSKBとを使用し、実際の中継判定時には第2サーチキーSKBと第3サーチキーSKCとを使用する。

【0046】ここで、本発明のネットワーク間中継先判定用データベースを使用した中継先判定の手順において使用されるサーチキーについて説明する。図7は図5に示されている各端末に関して、それぞれのMACアドレスと、それぞれが接続するブリッジBRのアドレス(ポートアドレス)と、それぞれが属するV-LANのアドレス(V-LANアドレス)とを一覧にして示した模式図である。第1サーチキーSKAとしては、4ビットのポートアドレス(但し、学習時には受信ポートアドレス)が、第2サーチキーSKBとしては48ビットのMACアドレス(但し、学習時には受信フレームのソースアドレス(SA)、実際の中継先判定時には受信フレームのデスティネーションアドレス(DA))が、第3サーチキーSKCとしては4ビットのV-LANアドレスがそれぞれ使用される。以下、図5に示されているネットワークの例について具体的に説明する。

【0047】図7に示されているように、ブリッジBRには各端末に関する情報として、それぞれのMACアドレスと、それぞれが属するV-LANアドレスと、ブリッジBRから見た各端末が接続されているポートアドレスとが与えられる。ここで、上述のように、ポートアドレスを第1サーチキーSKAに、各端末のMACアドレスを第2サーチキーSKBに、V-LANアドレスを第3サーチキーSKCにそ

れぞれ割り当てる。そして、第1ハッシュ関数HA(X)として、第1サーチキーSKA(ポートアドレス)の下位1ビットと第2サーチキーSKB(MACアドレス)の下位2ビットとをマージした値を、第2ハッシュ関数HB(X)として、第3サーチキーSKC(LANアドレス)の下位2ビットと第2サーチキーSKB(MACアドレス)の下位2ビットとをマージした値を採用する。この結果、それぞれの端末に関する第1ハッシュ関数HA(X)及び第2ハッシュ関数HB(X)は図8の模式図に示されているようになる。

【0048】従って、48ビットのMACアドレスと4ビットのポートアドレスと4ビットのV-LANアドレスとから、3ビットの第1ハッシュ関数HA(X)と4ビットの第2ハッシュ関数HB(X)とが生成される。第1ハッシュ関数HA(X)は3ビットであるため8通りしかなく、また第2ハッシュ関数HB(X)は4ビットであるため16通りしかない。

【0049】いまたとえば、図5に示されているV-LAN A内のMACアドレスが〔・・・0010〕である端末TFから同じくV-LAN A内のMACアドレスが〔・・・0000〕である端末TAへ通信を行なう場合について考える。この場合の送信フレームの図2の模式図に示されているヘッダ内のフィールドF1にはデスティネーションアドレスDAとして端末TAのMACアドレス〔・・・0000〕が、フィールドF2にはソースアドレスSAとして端末TFのMACアドレス〔・・・0010〕が入っている。またこの送信フレームは、端末TFから送出されてポートアドレス〔0000〕の第1ポートからブリッジBRに入力する。

【0050】このようにブリッジBRに端末TFからの送信フレームが入力した場合の学習時の処理手順を図9のフローチャート及び第1、第2ハッシュ関数と各エントリとのリンクの関係を示す図10の模式図を参照して説明する。

【0051】なお、本発明のネットワーク間中継先判定用データベースにおいては、学習時には第1ハッシュテーブルHTAを、実際の中継先のサーチ時には第2ハッシュテーブルHTBをそれぞれ使用する。両ハッシュテーブルHTA、HTBには、個々のエントリのポイントがハッシュ値をインデクスとして登録されている。従って、図10に示されているように、たとえばあるMACアドレスのハッシュ値“010”に対応する第1ハッシュテーブルHTAの位置にポイント(たとえばP1)が登録されている場合には、このポイント(P1)で指定される、即ちリンクされているエントリ(たとえばエントリE1)がまず最初に参照される。具体的には以下のようになる。

【0052】まず、端末TFから端末TAへの送信フレームがブリッジBRに受信されると(ステップS11)、その送信フレームがブリッジBRに入力してきたポート(この場合は第1ポートPT1)が検出されてそのポートアドレス〔0000〕が第1サーチキーSKAとされ、またその送信フ

フレームのヘッダからデスティネーションアドレスDA（端末TAのMACアドレス）〔・・・0000〕及びソースアドレスSA（端末TFのMACアドレス）〔・・・0010〕が取り出される。そして、第1ハッシュ関数HA(X)により第1サーチキーSKAであるポートアドレス〔0000〕の下位1ビット”0”と第2サーチキーSKBであるソースアドレスSA〔・・・0010〕の下位2ビット”10”とがマージされてハッシュ値が求められる（ステップS12）。この場合のハッシュ値は”010”になる。

【0053】ハッシュ値が求まるとそれをインデクスとして学習時用の第1ハッシュテーブルHTAがサーチされる（ステップS13）。第1ハッシュテーブルHTAのインデクス”010”に対応する位置にポインタ（たとえばP1）が既に登録されていれば（ステップS14で”YES”）、そのポインタP1でリンクされているエントリ（図10の例ではE1）が参照される（ステップS15）。ところで、エントリE1には第1サーチキーSKA1と第2サーチキーSKB1とが記録されているので、これらとハッシュ値が作成された元の送信フレームのソースアドレスSA（この場合は端末TFのMACアドレス〔・・・0010〕）、即ち第2サーチキーSKBとそれがブリッジBRへ入力したポートのアドレス（この場合は第1ポートのアドレス〔0000〕）、即ち第1サーチキーSKAとが比較され（ステップS16）、一致していればこのエントリE1が目的のエントリである（ステップS17で”YES”）。

【0054】しかし、上述のステップS16での比較結果が不一致であった場合（ステップS17で”NO”）にはそのエントリE1は目的のエントリではない。この場合、エントリE1の第1ネクストポインタNPA1にポインタ値が登録されているか否かが調べられ（ステップS18）、登録されていればそのポインタ値（図10の例ではP3）でリンクされているエントリ（図10の例ではE3）が次に参照される（ステップS15）。そして、前述同様にステップS16での処理が行なわれてこのエントリE3が目的のエントリであるか否かが調べられる。このエントリE3が目的のエントリでない場合（ステップS17で”NO”）には更にこのエントリE3の第1ネクストポインタNPA3に登録されているポインタ値（たとえばP6）でリンクされているエントリが参照される。

【0055】以下同様にして、目的のエントリ、即ちそのエントリの第1サーチキーSKA及び第2サーチキーSKBの値が送信フレームのソースアドレスSAとポートアドレスの値と一致するエントリが見つかるまで、チェイニングリストが順次的にサーチされるが、その間に第1ネクストポインタNPAに次のポインタ値が登録されていないエントリがあればそこでエントリのリンクが終了していることを意味している（ステップS18で”NO”）。従って、リンクしているエントリを順にたどっていても目的のエントリが見つからないままにリンクが終了した

場合には、対応するエントリが未だ登録されていないことを意味する。そのような場合にはMACアドレスに対応するエントリを新たに登録する必要があるが（ステップS20）、エントリの新規登録に関しては後述する。

【0056】このようにして目的のエントリが見つかった場合には、そのエントリに登録されているV-LANアドレス、即ち第3サーチキーSKCが取り出され（ステップS19）、それが中継先を判定する再のサーチキーとして使用される。またこの際、付属情報R内の更新履歴が更新される。

【0057】次に、上述のステップS20において実行される新規エントリの追加処理について、その手順を示す図11のフローチャートを参照して説明する。

【0058】ブリッジBRへ送信されてきたフレームのソースアドレスSAに対応するエントリが第1ハッシュテーブルHTAに登録されていない場合、即ちこの例では第1ハッシュテーブルHTAのインデクス”010”に対応する位置にポインタが登録されていない場合（ステップS14で”NO”）、またはポインタが登録されていても前述のようにそれにリンクされているいずれのエントリの第1サーチキーSKA及び第2サーチキーSKBの値も送信フレームのソースアドレスSA及びポートアドレスの値に一致しない場合（ステップS17及びS18で共に”NO”）には新規にエントリを追加する必要がある（ステップS20）。

【0059】まず新規のエントリが確保され（ステップS21）、第1ハッシュテーブルHTAのインデクスに対応する位置にポインタが未登録である場合（ステップS22で”NO”）には、新規エントリとリンクさせるためのポインタが第1ハッシュテーブルHTAの上述の例ではインデクス”010”に対応する位置に登録される（ステップS23）。一方、第1ハッシュテーブルHTAのインデクスに対応する位置にポインタが既に登録されているが、それにリンクするチェイニングリスト中に対応するエントリがない場合（ステップS22で”YES”）には、新規エントリとリンクするポインタがステップS18の処理で”NO”になったエントリ、即ちチェイニングリストの最後のエントリの第1ネクストポインタNPAに登録される（ステップS24）。

【0060】この後は新規エントリに対して、送信フレームのデスティネーションアドレス(DA)を第2サーチキーSKBとし、同じくV-LANアドレスを第3サーチキーSKCとして第2ハッシュ関数HB(X)からハッシュ値を求め、これを第2ハッシュテーブルHTB用の新たなインデクスとする（ステップS25）。この際、この新たなインデクスに対応する第2ハッシュテーブルHTBの位置にポインタが未だ登録されていなければその位置に新規エントリをリンクさせるためのポインタを登録する。一方、上述の新たなインデクスに対応する第2ハッシュテーブルHTBの位置にポインタが既に登録されていればそれによ

りリンクされているチェイニングリストの最後のエントリの第2ネクストポイントNPBに新規エントリをリンクさせるためのポイントを登録する(ステップS26)。

【0061】なお、新規エントリに第1サーチキーSKA、第2サーチキーSKB、第3サーチキーSKC等が登録されることは言うまでもない。

【0062】次に、実際の中継先判定時の処理について、図12に示されているフローチャートを参照して説明する。但し、上述の学習時の処理において送信フレームのソースアドレス(SA)に対応するエントリがなかった場合には、その送信フレームは廃棄されるか、または通信情報を調べることによって特定のV-LANに帰属させられる。このV-LANアドレスが中継先判定の際のサーチキー(第3サーチキーSKC)として使用される。しかし、送信フレームのソースアドレス(SA)に対応するエントリがある場合には、以下のようにして送信先ポートを決定する。

【0063】まず、送信フレームのデスティネーションアドレスDAを第2サーチキーSKBとし、上述の学習時の処理で取り出したV-LANアドレスを第3サーチキーSKCとして第2ハッシュテーブルHTBがサーチされる。具体的には、送信フレームのデスティネーションアドレスDA、即ち第3サーチキーSKCである端末TAが属するV-LANアドレス「0000」の下位2ビット「00」と、第2サーチキーSKBである端末TAのMACアドレス「・・・0000」の下位2ビット「00」とが第2ハッシュ関数HB(X)によりマージされてハッシュ値(この場合は「0000」)が求められる(ステップS31)。このハッシュ値「0000」が求まるとそれをインデクスとして第2ハッシュテーブルHTBが参照される(ステップS32)。第2ハッシュテーブルHTBのインデクス「0000」に対応する位置にポイント(図10の例ではP2)が既に登録されていれば(ステップS33で「YES」)、そのポイントP2にリンクしているエントリ(図10の例ではE2)を参照し(ステップS34)、そのエントリE2の第2サーチキーSKB2及び第3サーチキーSKC2の値と送信フレームのデスティネーションアドレス(DA)及びV-LANアドレスの値とを比較する(ステップS35)。

【0064】このステップS35での比較の結果が一致した場合(ステップS36で「YES」)であれば、そのエントリE2が目的のエントリであるので、そのエントリE2の第1サーチキーSKA2に登録されている値(図10の例ではX8)を取り出す(ステップS38)。この第1サーチキーSKA2の値X8が送信フレームの送出先ポート、即ち送信フレームのヘッダのデスティネーションアドレスで指定されている端末(この場合は端末TA)への送信フレームの送出先のポートのポートアドレスである。(ステップS35)。

【0065】一方、上述のステップS35での比較結果が不一致であった場合(ステップS36で「NO」)、そのエ

ントリE2に第2ネクストポイントNPB2が既に登録されているか否かが調べられ(ステップS37)、登録されていれば(ステップS37で「YES」)その値(図10の例ではP4)でリンクされているエントリ(図10の例ではE4)が参照される(ステップS34)。以下、前述の学習時の場合と同様にして、ステップS35での処理によりチェイニングリストが、目的のエントリが見つかるまで、またはリンクが終了するまで(第2ネクストポイントNPBが未登録であるエントリに到達するまで)順次サーチされる。そして、目的のエントリが見つければ、そのエントリの第1サーチキーSKAが取り出されてそれが送信フレームの送出先ポートとされ(ステップS38)、目的のエントリが見つからなかった場合にはサーチは終了し、そのV-LANに属するすべてのポート(但し、受信ポート、この場合はポートアドレス「0000」の第1ポートは除く)を送出先ポートと見做す。以上が実際の中継先判定時の処理である。

【0066】ところで、前述の図11のフローチャートに示されている新規エントリの追加に際しては、新規のエントリを単にチェイニングリストの最後のエントリにリンクさせるようにしているが、リンクの順序を工夫することにより、チェイニングリストのサーチを効率的に行なえるようにすることも可能である。

【0067】たとえば、図13の模式図に示されているように、それぞれポイント値P1、P2、P3、P4で特定されるエントリE1、E2、E3、E4があり、第1ハッシュテーブルHTAのインデクス「010」にポイント値P1が、各エントリE1、E2、E3、E4の第1ネクストポイントNPA1、NPA2、NPA3、NPA4にそれぞれポイント値P2、P3、P4、0(ポイント値が0である場合はそのエントリでリンクが終了する)が登録されているとすると、これらのエントリE1、E2、E3、E4はこの順にチェイニングされており、しかもエントリE4が最後のエントリであるということになる。更に、各エントリE1、E2、E3、E4の第1サーチキーSKA1、SKA2、SKA3、SKA4の値X11、X12、X13、X14と第2サーチキーSKB1、SKB2、SKB3、SKB4の値X21、X22、X23、X24とのマージ値(M1、M2、M3、M4で表す)の実際の値がそれぞれX1、X2、X3、X4であり、それらの間の大小関係がX1 < X2 < X3 < X4であるとする。

【0068】なお、図13においては、リンクされている各エントリE1、E2、E3、E4が実際にリンクして配列されているように示されているが、これはあくまでも説明の便宜のためであって、実際にはハードウェアとしての記憶装置内に散在して記憶されている。

【0069】このようなチェイニングリストが既に存在する場合に、たとえばポートアドレスの値がX1n、ソースアドレスSAの値がX2nであり、第1ハッシュテーブルHTAのインデクスが「010」になる未登録の端末からの送信フレームがブリッジBRに入力されたとすると、以下のようなサーチが行なわれ、新規エントリが追加され

る。但し、ポートアドレスの値 $X1n$ とソースアドレスSAの値 $X2n$ とのマージ値 $Mn$ の実際の $Xn$ (但しここで、 $X2 < Xn < X3$ であるとする)であるとする。

【0070】まず、第1ハッシュテーブルHTAのハッシュ値“010”に対応する位置に登録されているポインタ値P1が取り出され、このポインタ値P1にリンクされているエントリE1が参照される。エントリE1の第1サーチキーSKA1に登録されている値と第2サーチキーSKB1に登録されている値とのマージ値M1は $X1$ であり、送信フレームのソースアドレスSAとポートアドレスとのマージ値 $Mn$ である $Xn$ とは一致せず、しかも小さいので、次にエントリE1の第1ネクストポインタNPA1に登録されているポインタ値P2でリンクされているエントリE2が参照される。

【0071】エントリE2の第1サーチキーSKA2に登録されている値と第2サーチキーSKB2に登録されている値とのマージ値M2は $X2$ であり、これも送信フレームのソースアドレスSAとポートアドレスとのマージ値 $Mn$ である $Xn$ とは一致せず、しかも小さいので、次にエントリE2の第1ネクストポインタNPA2に登録されているポインタ値P3でリンクされているエントリE3が参照される。

【0072】エントリE3の第1サーチキーSKA3に登録されている値と第2サーチキーSKB3に登録されている値とのマージ値M3は $X3$ であり、これも送信フレームのソースアドレスとポートアドレスとのマージ値である $Xn$ とは一致せず、しかも今回は大きいので、対応するエントリが未登録であることが判明するためここでサーチは打ち切れ、新規エントリEnが追加される。

【0073】新規エントリEnは、図14の模式図に示されているように、チェイニングリスト中のエントリE2の次で且つエントリE3の前の順序になるように新たに各エントリがリンクされる。具体的には、新規エントリEnをリンクするためのポインタ値を $Pn$ とすると、エントリE2の第1ネクストポインタNPA2には新たにポインタ値 $Pn$ が登録され、新規エントリEnの第1ネクストポインタNPA $n$ にはエントリE3をリンクするためのポインタ値P3が登録され、また第1サーチキーSKA $n$ 及び第2サーチキーSKB $n$ にはそれぞれ送信フレームのポートアドレスの値である $X1n$ 及びソースアドレスSAの値である $X2n$ が登録される。

【0074】このように、新規エントリEnをチェイニングリストの最後(図13に示されている例ではエントリE4の次順)に単純にリンクする従来の手法とは異なり、サーチキーから一意に定められる値(上述の例では第1サーチキーSKAと第2サーチキーSKBとの値をマージした値)の順序に従って(上述の例では昇順であるが、降順でもよい)にエントリ相互をリンクさせておくことにより、チェイニングリストの全てのエントリをサーチせずとも目的のエントリが登録されていないことが迅速に判明する。

【0075】チェイニングリストのサーチを効率的に行なう目的で、更に他の手法を採用することも可能であ

る。

【0076】たとえば、図15の模式図に示されているように、前述の図13に示されている場合と同様のチェイニングリストに対して、第1ハッシュテーブルHTAには各インデクスに対応して二つのポインタ値が登録されている。なお、各エントリE1, E2, E3, E4の第1ネクストポインタNPA1, NPA2, NPA3, NPA4にもそれぞれ二つのポインタ値が登録されている。

【0077】第1ハッシュテーブルHTAには、各インデクスにリンクするチェイニングリストの先頭のエントリ、即ちそのエントリの第2サーチキーSKB(MACアドレス)の値が最小のエントリをリンクさせるための昇順ポインタIP(図15に示されている例ではその値はP1)と、末尾のエントリ、即ちそのエントリの第2サーチキーSKBの値が最大のエントリをリンクさせるための降順ポインタDP(図15に示されている例ではその値はP4)が登録されている。

【0078】一方、各エントリエントリE1, E2, E3, E4には前述の第1ハッシュテーブルHTAの昇順のポインタIPと降順のポインタDPとに対応して、第1昇順ネクストポインタINPAとしてチェイニングリストを昇順に見た場合に次順のエントリをリンクさせるためのポインタ値と、第1降順ネクストポインタDNPAとしてチェイニングリストを降順に見た場合に次順のエントリをリンクさせるためのポインタ値とが登録されている。たとえば、エントリE1の第1昇順ネクストポインタINPA1のポインタ値としてP2が、第1降順ネクストポインタDNPA1のポインタ値として0(ポインタ値が0である場合はそのエントリでリンクが終了する)が登録されている。また、エントリE2の第1昇順ネクストポインタINPA2のポインタ値としてはP3が、第1降順ネクストポインタDNPA2のポインタ値としてはP1が登録されている。また、エントリE3の第1昇順ネクストポインタINPA3のポインタ値としてはP4が、第1降順ネクストポインタDNPA3のポインタ値としてはP2が登録されている。更に、エントリE4の第1昇順ネクストポインタINPA4のポインタ値としては0が、第1降順ネクストポインタDNPA4のポインタ値としてはP3が登録されている。

【0079】この図15に示されているようなチェイニングリストが既に存在する場合に、たとえばソースアドレスSAの値が $X2$ であり、第1ハッシュテーブルHTAのインデクスが“010”になる未登録の端末からの送信フレームがブリッジBRに入力されたとなると、以下のようなサーチが行なわれる。

【0080】まず、第1ハッシュテーブルHTAのハッシュ値“010”に対応する位置に登録されている昇順ポインタIPのポインタ値P1と降順ポインタDPのポインタ値P4との内から送信フレームのソースアドレスSAの値 $X2$ に応じていずれかが選択される。この選択は厳密に行なう必要はなく、たとえば送信フレームのソースアドレスSA

の値がとり得る値の範囲を二分したいずれに属するかで決定すればよい。

【0081】たとえば、昇順のポインタ値P1が選択された場合には、このポインタ値P1にリンクされているエントリE1が参照される。エントリE1の第2サーチキーSKB1に登録されている値X1は送信フレームのソースアドレスSAの値X2とは一致しないので、次にエントリE1の第1昇順ネクストポインタDNPA1に登録されているポインタ値P2でリンクされているエントリE2が参照される。エントリE2の第2サーチキーSKB2に登録されている値X2は送信フレームのソースアドレスSAの値X2と一致するため、このエントリE2が目的のエントリであることが判明し、ここでサーチは打ち切られる。

【0082】一方、第1ハッシュテーブルHTAにおいて降順のポインタ値P4が選択された場合には、このポインタ値P4にリンクされているエントリE4が参照される。エントリE4の第2サーチキーSKB4に登録されている値X4は送信フレームのソースアドレスSAの値X2とは一致しないので、次にエントリE4の第1降順ネクストポインタDNPA4に登録されているポインタ値P3でリンクされているエントリE3が参照される。このエントリE3の第2サーチキーSKB3に登録されている値X3は送信フレームのソースアドレスSAの値X2とは一致しないので、次にエントリE3の第1降順ネクストポインタDNPA3に登録されているポインタ値P2でリンクされているエントリE2が参照される。エントリE2の第2サーチキーSKB2に登録されている値X2は送信フレームのソースアドレスSAの値X2と一致するため、このエントリE2が目的のエントリであることが判明し、ここでサーチは打ち切られる。

【0083】なお、新規エントリの登録は基本的には前述の図14に示されている例と同様にして行なわれる。但し、新規エントリにも第1昇順ネクストポインタINPAと第1降順ネクストポインタDNPAとが登録される。また、前述の図13に示されている手法とこの図15に示されている手法とを組み合わせることも勿論可能であり、その場合には目的のエントリが既に登録されている場合には従来に比してより迅速に目的のエントリを見つけることが可能になるのみならず、目的のエントリが未登録である場合にはその目的のエントリが登録されていないことが迅速に判明する。

【0084】チェイニングリストのサーチを効率的に行なう目的で、更に他の手法を採用することも可能である。

【0085】たとえば、図16の模式図に示されているように、前述の図15に示されている場合と同様のチェイニングリストに対して、第1ハッシュテーブルHTAには各インデクスに対応してポインタPと、その値でリンクされているエントリの識別コードIDの値とが登録されている。識別コードIDは具体的には、対応するエントリに登録されている第1サーチキーSKAと第2サーチキーSKB

とのマージ値Mであり、個々の端末に1対1で対応する。図16に示されている例では具体的には、第1ハッシュテーブルHTAのインデクス"010"に対応する位置にはチェイニングリストのほぼ中央にリンクされているエントリE2（図16に示されている例ではE1, E2, E3, E4の4個のエントリの内の昇順で2番目）をリンクさせるためのポインタ値P1と、エントリE2に登録されている第1サーチキーSKAと第2サーチキーSKBとのマージ値M2の値X2が識別コードIDとして登録されている。

【0086】なお、各エントリE1, E2, E3, E4には前述の図15に示されている例と同様に、第1昇順ネクストポインタINPAとしてチェイニングリストを昇順に見た場合に次順のエントリをリンクさせるためのポインタ値と、第1降順ネクストポインタDNPAとしてチェイニングリストを降順に見た場合に次順のエントリをリンクさせるためのポインタ値とが登録されている。なお、図示されていないが、第2昇順ネクストポインタINPB及び第2降順ネクストポインタDNPBも各エントリE1, E2, E3, E4に登録されており、それらは第2ハッシュテーブルHTBに対応して備えられている。

【0087】この図16に示されているようなチェイニングリストが既に存在する場合に、たとえばソースアドレスSAとポートアドレスとのマージ値MnがX1であり、第1ハッシュテーブルHTAのインデクスが"010"になる未登録の端末からの送信フレームがブリッジBRに入力されたとすると、以下のようなサーチが行なわれる。

【0088】まず、第1ハッシュテーブルHTAのハッシュ値"010"に対応する位置に登録されているエントリE2の第1サーチキーSKA1と第2サーチキーSKB1の値が読み出され、それらのマージ値M2であるX2が送信フレームのソースアドレスSAとポートアドレスとをマージした値X1と比較される。この場合、送信フレームのソースSAとポートアドレスとの値のマージ値M1である値X1は第1ハッシュテーブルHTAに登録されていた識別コードIDの値X2よりも小さいので、第1ハッシュテーブルHTAに登録されていたポインタPのポインタ値P1にリンクされているエントリE2がまず参照され、爾後は降順にチェイニングリストがサーチされる。

【0089】具体的には、まずポインタ値P2でリンクされているエントリE2が参照される。エントリE2に登録されている第1サーチキーSKA2と第2サーチキーSKB2との値のマージ値M2であるX2は送信フレームのソースアドレスとポートアドレスとをマージした値X1とは一致しないので、次にエントリE2の第1降順ネクストポインタDNPA2に登録されているポインタ値P1でリンクされているエントリE1が参照される。エントリE1に登録されている第1サーチキーSKA1と第2サーチキーSKB1との値のマージ値M1であるX1は送信フレームのソースアドレスとポートアドレスとをマージした値X1と一致するため、このエントリE1が目的のエントリであることが判明し、ここで

サーチは打ち切られる。

【0090】なお、送信フレームのソースアドレスSAとポートアドレスとのマージ値がたとえばX3である場合には、第1ハッシュテーブルHTAのインデックス”010”の位置に登録されている識別コードIDの値X2よりも大きいため、エントリE2が最初に参照され、爾後は昇順にチェイニングリストがサーチされる。その手順は既に説明した例と同様であるので省略する。

【0091】なお、新規エントリの登録は基本的には前述の図14に示されている例と同様にして行なわれる。但し、新規エントリにも第1昇順ネクストポイントINPAと第1降順ネクストポイントDNPAとが登録される。また、前述の図13に示されている手法とこの図16に示されている手法とを組み合わせることも勿論可能であり、その場合には目的のエントリが既に登録されている場合には従来に比してほぼ半数のエントリのサーチで目的のエントリを見つけることが可能になるのみならず、目的のエントリが未登録である場合にはその目的のエントリが登録されていないことが迅速に判明する。

【0092】なお、上述のエントリのサーチを効率的に行なうための手法は、第1ハッシュテーブルHTAとのリンクに関して説明したが、各エントリには送信フレームのデスティネーションアドレスとV-LANアドレスとも登録されているため、第2ハッシュテーブルHTBとのリンク関係においても適用可能であることは言うまでもない。また更に、上述のエントリのサーチを効率的に行なうための手法は、本発明が対象とするネットワーク間中継先判定用データベースのような各エントリが二つのハッシュテーブルとリンクするような場合のみならず、従来の単純なハッシュテーブルとエントリとのリンク関係

においても適用可能であることは言うまでもない。

【0093】

【発明の効果】以上に詳述したように、本発明に係るネットワーク間中継先判定用データベースによれば、ハッシュ関数を使用することにより本来のサーチキーをサーチするために必要な空間を圧縮することは従来と同様であるが、学習時と実際の中継先判定時とで異なるハッシュテーブルを使用し、しかも各エントリに両方のハッシュテーブル用のポイントを持たせておくことにより、両ハッシュテーブルにリンクするエントリを共通化することにより、エントリ数を従来のハッシュテーブル使用の場合と同等に抑えることが可能になる。

【0094】また、各エントリのリンクの順序を端末のアドレス、ポートのアドレス、ネットワークのアドレスに応じて工夫することにより、目的のエントリを迅速にサーチし、また目的のエントリが未登録であることを迅速に検出し得るようになる。なお、このような手法は本発明のネットワーク間中継先判定用データベースのみならず、一般のデータベースの構築に際しても採用可能であり、サーチキーに対応して各エントリのリンクの順序

を工夫することにより、目的のエントリを迅速にサーチし、また目的のエントリが未登録であることを迅速に検出し得るようになる。

【図面の簡単な説明】

【図1】ブリッジを使用したLAN間接続の一般的な例を示す模式図である。

【図2】LAN間ネットワークにおいて送信される信号の一般的なフォーマットを示す模式図である。

【図3】従来の一般的なハッシュテーブル及び個々のエントリの関係を示す模式図である。

【図4】LAN間接続されている各端末のMACアドレスの例を示す模式図である。

【図5】本発明が適用されるブリッジを使用したLAN間接続の例を示す模式図である。

【図6】本発明のネットワーク間中継先判定用データベースにおける個々のエントリの内容を示す模式図である。

【図7】各端末に関して、それぞれのMACアドレスと、それぞれが接続するブリッジのアドレス（ポートアドレス）と、それぞれが属するV-LANのアドレス（V-LANアドレス）とを一覧にして示した模式図である。

【図8】それぞれの端末に関する第1ハッシュ関数HA(X)及び第2ハッシュ関数HB(X)を示す模式図である。

【図9】本発明のネットワーク間中継先判定用データベースによる学習時の処理手順を示すフローチャートである。

【図10】本発明のネットワーク間中継先判定用データベースによる学習時の第1、第2ハッシュ関数と各エントリとのリンクの関係を示す模式図である。

【図11】本発明のネットワーク間中継先判定用データベースによる学習時の新規エントリの追加処理の手順を示すフローチャートである。

【図12】本発明のネットワーク間中継先判定用データベースによる中継先判定時の処理手順を示すフローチャートである。

【図13】チェイニングリストのサーチを効率的に行なうための手法を説明するための模式図である。

【図14】チェイニングリストのサーチを効率的に行なうための手法を説明するための模式図である。

【図15】チェイニングリストのサーチを効率的に行なうための手法を説明するための模式図である。

【図16】チェイニングリストのサーチを効率的に行なうための手法を説明するための模式図である。

【符号の説明】

BR   ブリッジ（中継装置）

X, Y, Z   LAN

TA, TB, TC…   端末

F1   デスティネーションアドレス(DA)のフィールド

F2   ソースアドレス(SA)のフィールド

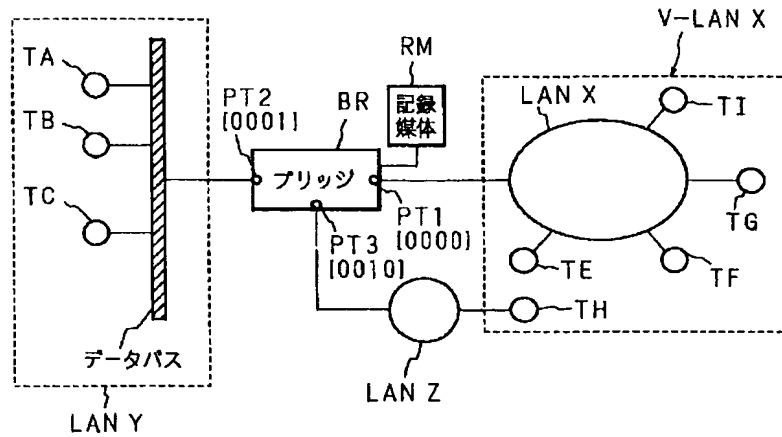
50   Ei   エントリ

Pi ポインタ  
 NPAi 第1ネクストポインタ  
 NPBi 第2ネクストポインタ  
 SKAi 第1サーチキー  
 SKBi 第2サーチキー

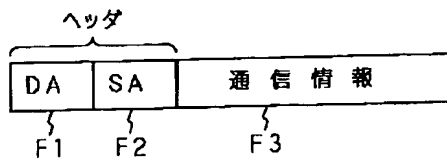
\* SKCi 第3サーチキー  
 ID 識別コード  
 HTA 第1ハッシュテーブル  
 HTB 第2ハッシュテーブル

\*

【図1】

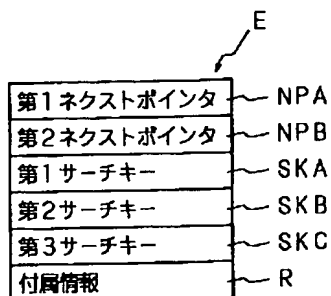


【図2】



DA: デスティネーション(送信先)アドレス  
 SA: ソース(送信元)アドレス

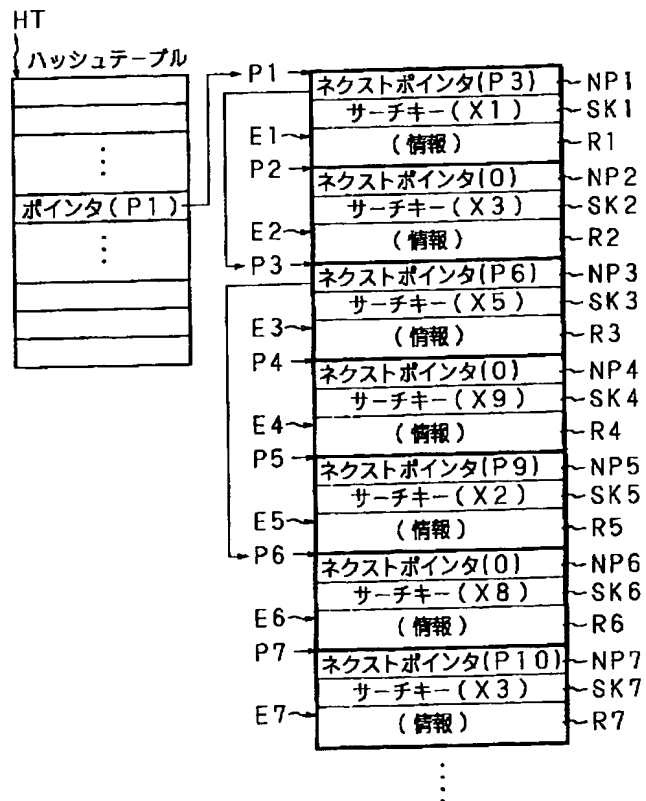
【図6】



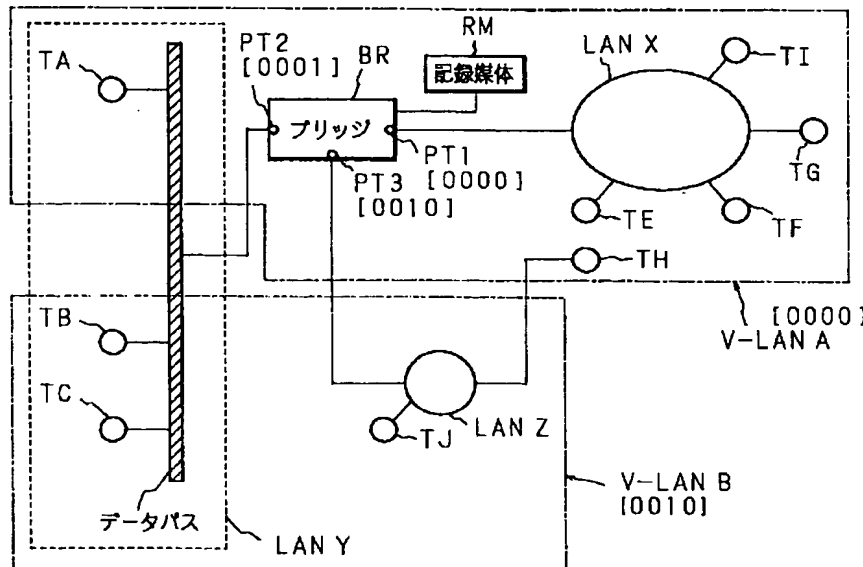
【図4】

端末	MACアドレス(48ビット)	ハッシュ関数(3ビット)
TA	0000	000
TB	0001	001
TC	0010	010
...	...	...
TE	0001	001
TF	0010	010
...	...	...

【図3】



【図5】



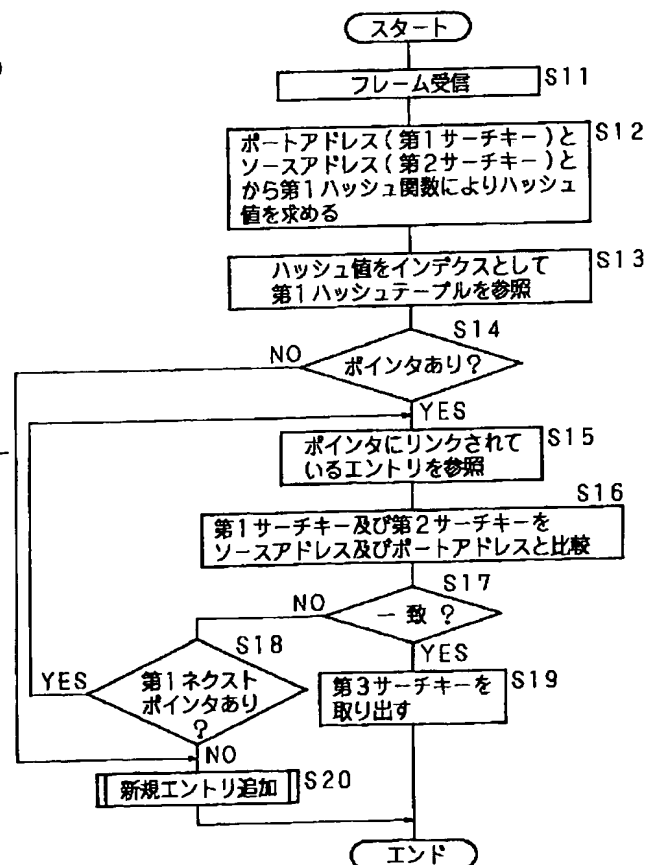
【図8】

端末	第1ハッシュ関数HA(X) (3ビット)	第2ハッシュ関数HB(X) (4ビット)
TA →	100	0000
TB →	101	1001
TC →	110	1010
⋮	⋮	⋮
TE →	001	0001
TF →	010	0010
⋮	⋮	⋮

【図7】

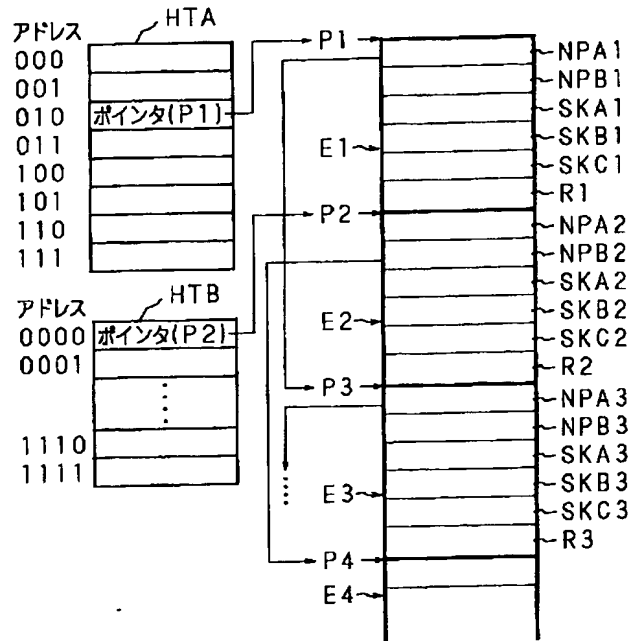
端末	MACアドレス(48ビット)	ポート アドレス (4ビット)	V-LAN アドレス (4ビット)
TA	0000	0001	0000
TB	0001	0001	0010
TC	0010	0001	0010
⋮	⋮	⋮	⋮
TE	0001	0000	0000
TF	0010	0000	0000
⋮	⋮	⋮	⋮
	第2サーチキー-SKB	第1サーチキー-SKA	第3サーチキー-SKC

【図9】



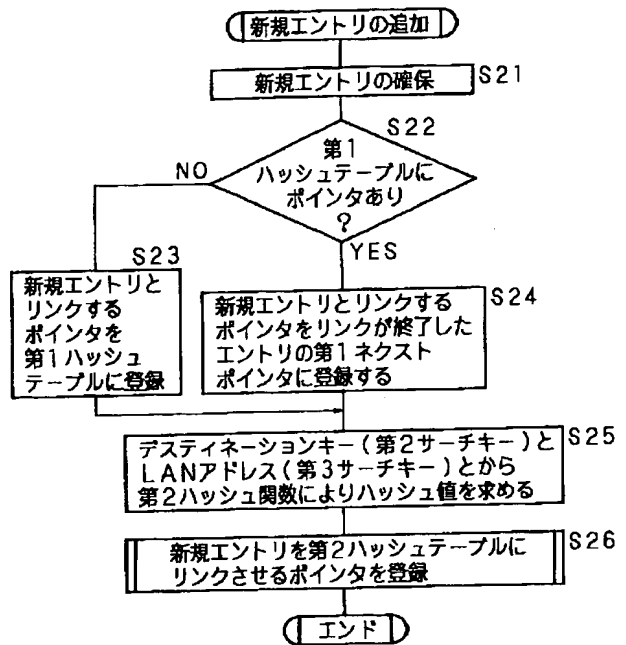


【図10】

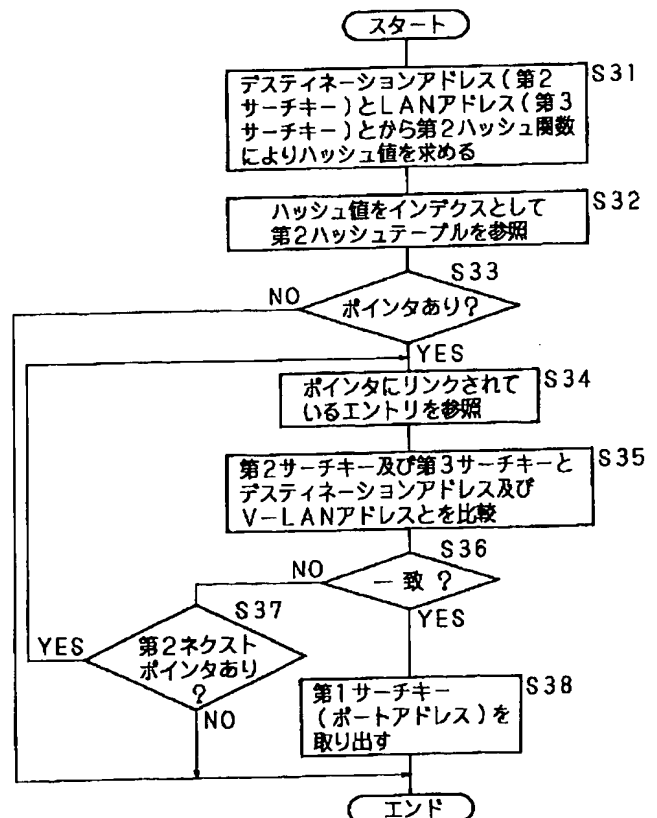


NPA1: 第1ネクスト  
ポインタ  
NPB1: 第2ネクスト  
ポインタ  
SKA1: 第1サーチキー  
SKB1: 第2サーチキー  
SKC1: 第3サーチキー

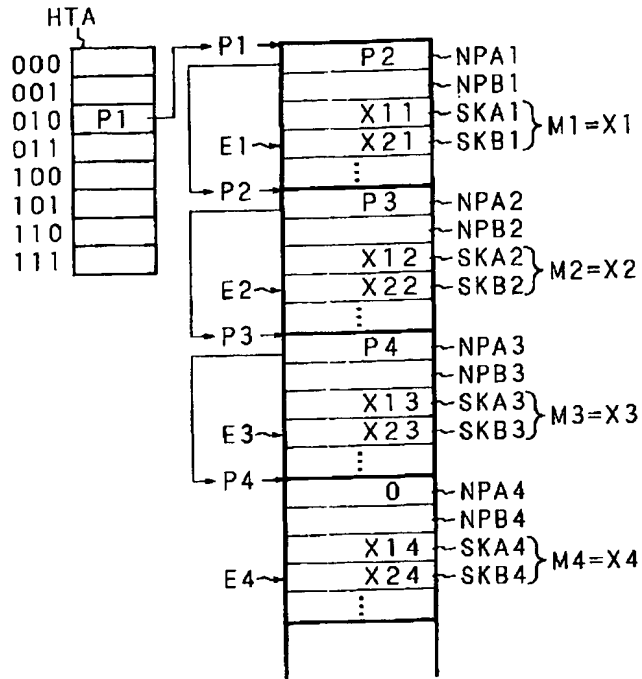
【図11】



【図12】

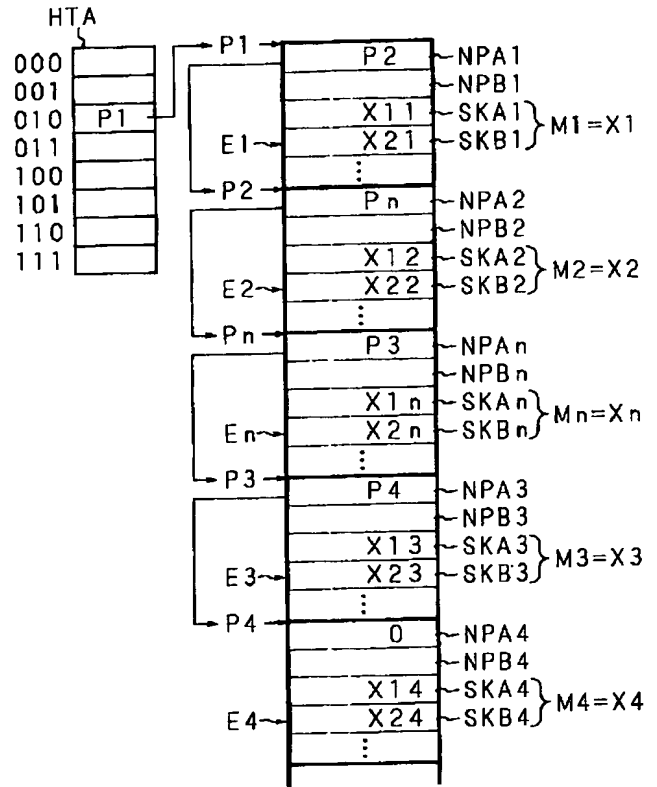


【図13】

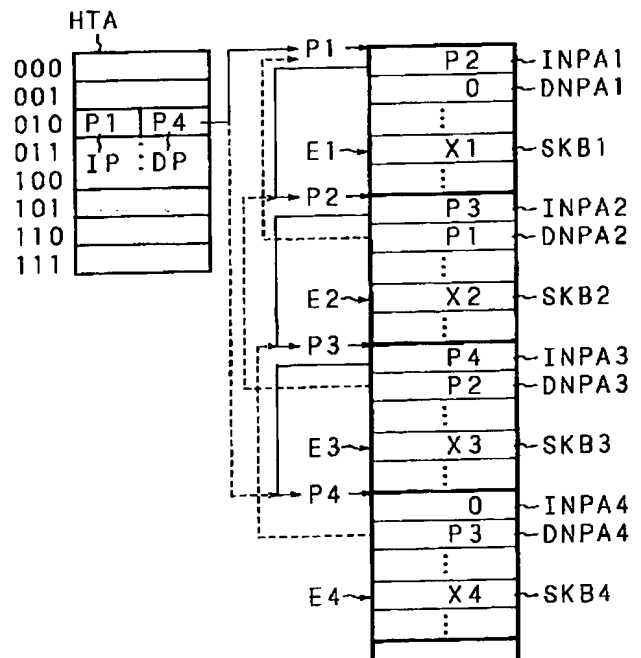


E1: エントリ  
 NPA1: 第1ネクストポイント  
 SKA1: 第1サーチキー  
 SKB1: 第2サーチキー  
 M1: 第1サーチキーと第2サーチキーの  
 値をマージした値

【図14】

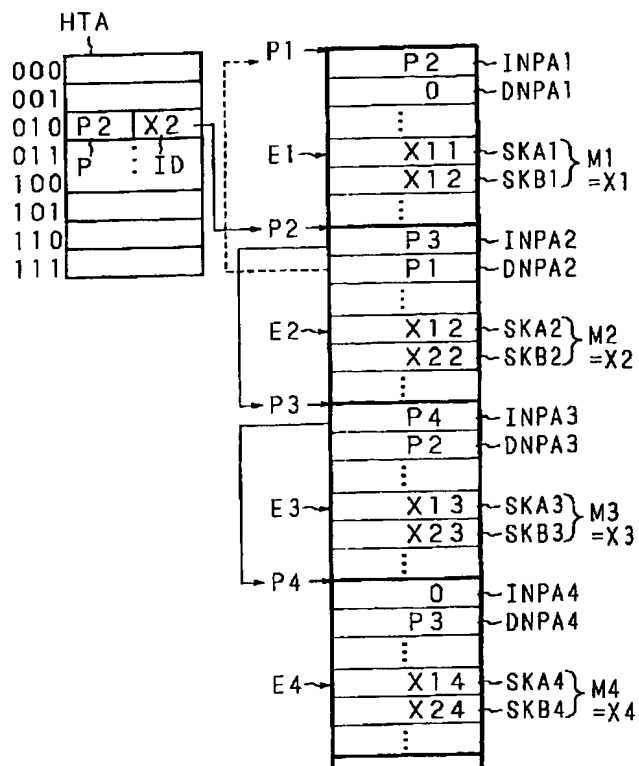


【図15】



E1: エントリ  
 INPA1: 第1昇順ネクストポイント  
 DNPA1: 第1降順ネクストポイント  
 SKB1: 第2サーチキー

【図16】



E i : エントリ  
 INPA i : 第 i 昇順ネクストポインタ  
 DNPA i : 第 i 降順ネクストポインタ  
 ID : 識別コード

**This Page is Inserted by IFW Indexing and Scanning  
Operations and is not part of the Official Record**

**BEST AVAILABLE IMAGES**

Defective images within this document are accurate representations of the original documents submitted by the applicant.

Defects in the images include but are not limited to the items checked:

- ☐ **BLACK BORDERS**
- ☐ **IMAGE CUT OFF AT TOP, BOTTOM OR SIDES**
- ☒ **FADED TEXT OR DRAWING**
- ☒ **BLURRED OR ILLEGIBLE TEXT OR DRAWING**
- ☐ **SKEWED/SLANTED IMAGES**
- ☐ **COLOR OR BLACK AND WHITE PHOTOGRAPHS**
- ☐ **GRAY SCALE DOCUMENTS**
- ☐ **LINES OR MARKS ON ORIGINAL DOCUMENT**
- ☐ **REFERENCE(S) OR EXHIBIT(S) SUBMITTED ARE POOR QUALITY**
- ☐ **OTHER:** \_\_\_\_\_

**IMAGES ARE BEST AVAILABLE COPY.**

**As rescanning these documents will not correct the image problems checked, please do not report these problems to the IFW Image Problem Mailbox.**